WHAT IS CLAIMED IS:

1		1.	A processing core that executes a compare instruction, the				
2	processing core comprising:						
3		a plura	ality of general-purpose registers comprising a first input operand				
4	register, a seco	ond inp	ut operand register and an output operand register;				
5		a regis	ter file comprising the plurality of general-purpose registers;				
6		compa	rison logic coupled to the register file, wherein the comparison logic				
7	tests for at leas	tests for at least two of the following relationships: less than, equal to, greater than and no					
8	valid relations	valid relationship;					
9		decode logic which selects the output operand register from the plurality of					
0	general-purpose registers; and						
1		a store path between the comparison logic and the selected output operand					
2	register.						
1		2.	The processing core that executes the compare instruction as set				
2		forth in claim 1, wherein a very long instruction word includes a plurality of compare					
3	instructions.						
1		3.	The processing core that executes the compare instruction as set				
2	forth in claim	1, wher	ein decode logic selects the first and second input operand registers				
3	from the plurality of general-purpose registers.						
1		4.	The processing core that executes the compare instruction as set				
2	forth in claim 1, wherein the processing core issues a plurality of compare instructions at						
3	one time.						
1		5.	The processing core that executes the compare instruction as set				
2	forth in claim	1, furth	er comprising:				
3		a first	load path between the first input operand register and comparison				
4	logic; and						
5		a secon	nd load path between the second input operand register and				
6	comparison logic.						
1		6.	The processing core that executes the compare instruction as set				
2	forth in claim	1, wher	ein the output operator register stores a value indicating a				

8

3	relationship b	relationship between the first and second input operator registers which is at least one of					
4	greater than, less than, equal to and not a number.						
1		7.	The second state of the se				
			The processing core that executes the compare instruction as set				
2		forth in claim 6, wherein the not a number value indicates a comparison between the first					
3	and second in	put ope	rands that cannot be made.				
1		8.	The processing core that executes the compare instruction as set				
2	forth in claim 6, wherein the value is an integer.						
1		9.	The processing core that executes the compare instruction as set				
2	forth in claim	laim 1, wherein:					
3		the first input operand register is a double precision floating point data					
4	type;	the In	so impart operand register is a dodole precision moating point data				
	type,	41					
5		the second input operand register is a single precision floating point data					
5	type; and						
7		the ou	tput operand register is a double precision floating point data type.				
1		10.	The processing core that executes the compare instruction as set				
2	forth in claim	m 1, further comprising a plurality of processing paths that are coupled to the					
3	register file.						
1		11.	The processing core that executes the compare instruction as set				
	Careta in alain		•				
2	forth in claim 1, wherein the register file comprises special purpose registers which						
3	cannot store a	n outpu	t operand.				
Į		12.	A method for performing a compare operation, the method				
2	comprising st	eps of:					
3		decodi	ing a compare instruction;				
1		config	uring first and second paths between a register file and comparison				
5	logic;						
ó		config	uring a third path between the comparison logic and the register file;				
7		compa	uring a first input operand and a second input operand to produce a				

result which indicates an absence of at least three mathematical relationships between the

first input operand and the second input operand; and

0	C	coupling an output operand to a general-purpose register in the register				
1	file.					
1 2	claim 12, the me	The method for performing the compare operation as set forth in hod further comprising a step of enabling the comparison logic in an				
3	arithmetic logic unit.					
1	1-	. The method for performing the compare operation as set forth in				
2	claim 12, wherei	erein the configuring steps each comprise a step of addressing a general-				
3	purpose register in the register file.					
1	1:	The method for performing the compare operation as set forth in				
2	claim 12, wherein a very long instruction word comprises the compare operation.					
1	10	The method for performing the compare operation as set forth in				
2	claim 12, wherei	wherein the comparing step comprises a step of converting a data type of at				
3	least one of the first and second input operands.					
1	11	A method for executing a compare instruction in a processor, the				
2	method comprisi	mprising steps of:				
3	is	issuing the compare instruction;				
4	co	mparing a first input operand and a second input operand to determine at				
5	least two mathen	least two mathematical relationships between the first and second input operands;				
6	de	ermining an output operand indicative of the mathematical				
7		relationships; and				
8	st	ring the output operand in a general-purpose register of a register file.				
1	18	The method for executing the compare instruction in the processor				
2	as set forth in cla	set forth in claim 17, wherein the comparing step comprises:				
3	de	ermining if the first input operand is less than the second input operand;				
4	de	ermining if the first input operand is greater than the second input				
5	operand;					
6	de	ermining if the first input operand is equal to the second input operand;				
7	and					
8	de	ermining if there is no valid relationship between the first input operand				
9	and the second input operand.					

. .

- 19. The method for executing the compare instruction in the processor as set forth in claim 17, wherein the compare instruction is a very long instruction word which comprises a plurality of compare instructions which are processed in parallel down separate processing paths.
- 20. The method for executing the compare instruction in the processor as set forth in claim 17, wherein the general-purpose register is used to store operators from other types of instructions.